Identification of Human Factors in Aviation Incidents Using a Data Stream Approach

Donghui Shi
Department of Computer Engineering
School of Electronics and Information Engineering
Anhui Jianzhu University
Hefei, China 230601
sdonghui@gmail.com

Jozef Zurada
Department of Computer Information Systems
College of Business
University of Louisville
Louisville, KY 40292
jozef.zurada@louisville.edu

Jian Guan
Department of Computer Information Systems
College of Business
University of Louisville
Louisville, KY 40292
jeff.guan@louisville.edu

Abstract

This paper investigates the use of data streaming analytics to better predict the presence of human factors in aviation incidents with new incident reports. As new incidents data become available, the fresh information can help not only evaluate but also improve existing models. First, we use four algorithms in batch learning to establish a baseline for comparison purposes. These are NaiveBayes (NB), Cost Sensitive Classifier (CSC), Hoeffdingtree (VFDT), and OzabagADWIN (OBA). The traditional measure of the classification accuracy rate is used to test their performance. The results show that among the four, NB and CSC are the best classification algorithms. Then we test the classifiers in a data stream setting. The two performance measure methods Holdout and Interleaved Test-Then-Train or Prequential are used in this setting. The Kappa statistic charts of Prequential measure with a sliding window show that NB exhibits the best performance, and is better than the other algorithms. The two different measure methods, batch learning with 10-fold cross validation and data stream with Prequential measure, get one consistent result. CSC is a suitable for unbalanced data in batch learning, but it is not best in Kappa statistic for data stream. Valid incremental algorithms need to be developed for the data stream with unbalanced labels.

1. Introduction

The Aviation Safety Reporting System (ASRS) is provided by the U.S. National Aviation Safety Data Analysis Center. It includes many confidential aviation incident reports, which are collected from volunteers, such as flight and ground crews. The goal of the ASRS is to enhance aviation safety by providing a venue where pilots, air traffic controllers, flight attendants, mechanics, ground personnel, and others involved in aviation operations can share information about unsafe situations that they have encountered or observed during flight or on the ground.

The reports contain numeric and textual data. A critical field in these reports is the textual description in each incident report. Reports are generated as data stream from airports in the U.S. every day. Because the percent of aviation mishaps caused by human errors is around 90 percent [18], correct identification of the presence of human factors in aviation incidents is a very important task [13], [18].

Posse et al. pointed out that machine learning techniques could extract information from the aviation safety reports automatically and reduce human involvement [16]. Two machine learning (ML) methods, classification and clustering, were used in identification of human factors in aviation incidents. Some studies used four classification algorithms to classify event types and provided some promising results. An expert was presented with one hundred reports categorized by event types using ML techniques. The correct rate that the expert agreed with the top-ranked choice is 73%. [4]. Péladeau et al. [15] looked for antonyms, synonyms, hypernyms, hyponyms, coordinate terms, homonyms, metonyms in the reports using a Wordnet based lexical database, next used clustering algorithms to show clusters graphically, and then grouped the words according to their co-occurrences. In a more recent study, Andrzejczak et al. used the Text Analytics feature in PASW Modeler 13 to link certain keywords in the reports to Skill–Rule–Knowledge (SRK) Taxonomy of Self-Reported Anomalies and constructed document webs examining strengths of associations of concept categories within records [2]. The concept categories which were extracted from aviation reports were Unsafe conditions, Rule-based errors, Skill-based errors, Knowledge-based errors, Weather, Aircraft issues, and Perceptual errors.

The above research used traditional batch learning algorithms and evaluation measures. In traditional batch learning, multiple models are constructed

URI: http://hdl.handle.net/10125/41280
ISBN: 978-0-9981331-0-2

through selecting training and test data randomly from a limited dataset and the final classification accuracy rate is obtained by averaging over the number of models created for different folds and runs. A k-fold cross-validation method is commonly used to evaluate the classification performance of the models. However, the main disadvantage of batch learning techniques is that they do not utilize the incremental incident data that accumulate in real time. The aviation incidents report analysis can potentially benefit from a data stream approach. Data stream learning algorithms can take snapshots at different times during the induction of a model to see how much the model improves or worsens over time [1], [3].

A data stream environment has different requirements from the traditional setting [5]. The most significant features are the following: 1) Process an example at a time, and inspect it only once; 2) Use a limited amount of memory; 3) Work in a limited amount of time; 4) Be ready to predict at any time. The process of stream learning algorithms is a repeated cycle [5]. The model constructed from initial data is constantly updated according to input cases from the stream. The algorithms execute in a limited amount of memory and within time bounds. A predictive model can be updated after processing a new input case. Common learning algorithms in stream scenarios are classification, clustering and outlier analysis. Recent studies in data stream systems show significant progress in the use of data stream methods in these areas. Examples of work include the research on data stream clustering algorithms [8], classification models for real estate data stream [12] and the use of Kappa statistic for evaluating time-changing Twitter data streams with unbalanced classes [6].

The aviation incident reports dataset presents unique challenges that can be addressed by data stream methods. The incident reports are somewhat unbalanced as about 1/3 of the records have been classified by human experts as caused by human related factors and the rest by non-human factors. In a traditional batch learning setting, oversampling and undersampling [19] can be used to alter the class distribution of the training data for an unbalanced data set. The disadvantage with undersampling is that it discards potentially useful data. The main disadvantage with oversampling is that by artificially making exact or very similar copies of existing cases, it makes overfitting likely. Another approach used to analyze unbalanced data sets in batch learning setting is cost-sensitive learning. It was used, for example, by Shi et al. [17] to recover bad debt in the healthcare industry. In this paper, we study the performance and evaluation measures of predictive models constructed from the data stream of aviation incidents with unbalanced class labels. The paper is organized as follows. Section 2 provides the data description and Section 3 presents topic mining, the data stream learning model and the evaluation measures used in the study. Section 4 discusses the results of the experiments. Finally, Section 5 concludes the paper and proposes some probable directions for the future research.

2. Data Description

The data were obtained from ASRS and consist of 18734 incident reports. Of these 18734 incidents about 1/3 have been classified by human experts as caused by human related factors and the rest by non-human factors [14]. Each incident report contains structured data fields such as time, place, environment, aircraft, component, personnel, events, etc. The unstructured textual data contain narratives provided by the flight and ground crews which pertain to the detailed description of the incidents. Only the textual data were used as the data stream and the structured numeric data were not used in building the models. The class variable is called Primary Factor and it may contain values representing different factors, including the human factor. We reassigned the values of the class variable Primary Factor to be either 1 for "human factor" or 0 for "non-human factor".

The approach used in this study consists of the following main processes: Natural Language Processing (NLP), topic mining, data stream modeling, and evaluation. NLP was used to parse and filter the narratives. The results from NLP were used to extract topics, which are taken as input variables of the data stream model. We initially selected 5000 examples, from the first record to the 5000th record of the data stream, as input cases. The data stream with 5000 records was used for all four scenarios.

3. Methodology

In the study, we used Massive Online Analysis (MOA) as the platform for data stream learning and evaluation measure. We used Weka 3.7 as the platform for batch learning. MOA is a system for online learning from data streams [1]. It is available in WEKA, an open-source machine learning software package. Figure 1 shows the architecture of identification of human factors in aviation incidents by using the traditional batch learning and learning from data streams.

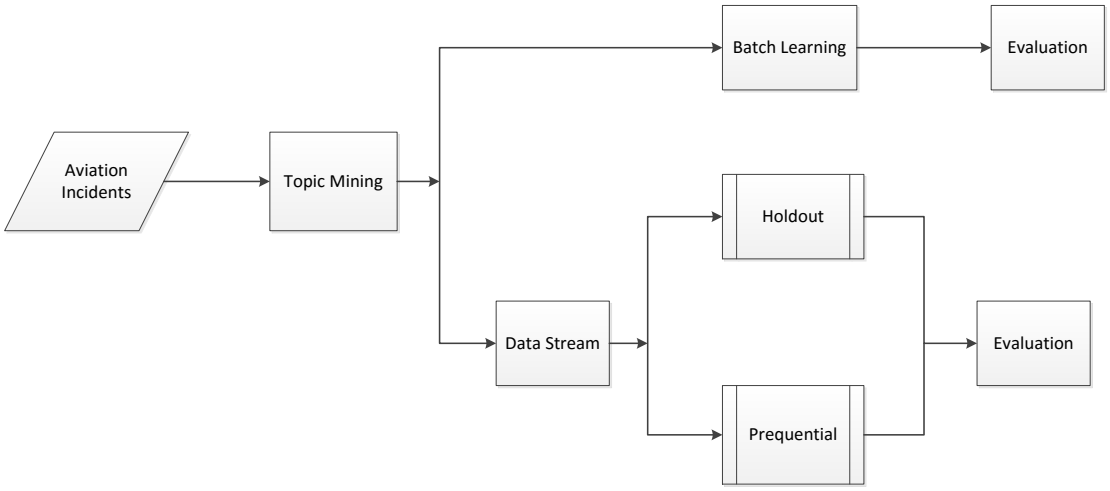

Figure 1. The architecture of identification of human factors in aviation incidents

3.1. Topic mining

Since the incident description in each report is the major input in our model, the first step was to transform the textual data of the description into a structured form. The first part of this transformation was the extraction of topics from the textual data using a Latent Semantic Analysis (LSA) method. These topics are in the form of top loaded terms and each report is assigned to a topic. There were two main steps in the topic mining process. The first step applied NLP technique to prepare the textual context for topic extraction. In the second step topics were extracted from the textual data. Because the aviation database uses a categorical target variable (Human factor or Non-human factor), the term weight function used was Mutual Information. In the Mutual Information weight function the weight is proportional to the similarity of the distribution of documents containing the term to the distribution of documents that are contained in the respective category. Term weights w_i are:

$$w_i = max_{c_k}\left[log\left(\frac{P(t_i, C_k)}{P(t_i)P(C_k)}\right)\right]$$

where $p(t_i)$ is the proportion of documents that contain term t_i, $p(c_k)$ is the proportion of documents that belong to category c_k, and $p(t_i, c_k)$ is the proportion of documents that contain term t_i belonging to category c_k. The number of topics extracted was the default value 25 in SAS Enterprise Miner 12.3, the tool used in this part of the implementation. Other numbers may be tested in search of a better classification performance. These topic assignments for the incident cases were then used as input for training the classifiers in the next step. The three examples of the topics are: 1) realize, look time, mistake, turn; 2) student, instructor, pattern, cessna, turn; and 3) factor, contribute, fatigue, time, miss.

3.2. Data stream classification algorithms

Four data stream algorithms were tested in our study and they are Naive Bayes (NB) [5], Hoeffding Tree (VFDT) [10], OzaBagADWIN (OBA) [5], and Cost Sensitive Classifier (CSC) [20]. NB performs classic Bayesian prediction while making the naive assumption that all inputs are independent. NB is a classification algorithm known for its simplicity and low computational cost. VFDT is an incremental, anytime decision tree induction algorithm that is capable of learning from data streams. VFDT exploits the fact that a small sample can often be enough to choose an optimal splitting attribute. This idea is supported mathematically by the Hoeffding bound. OBA is a change detector and estimator that solves in a well-specified way the problem of tracking the average of a stream of bits or real-valued numbers. The OBA classifier is based on the online bagging method of Oza and Rusell with the addition of the ADWIN algorithm as a change detector. When a change is detected, the worst classifier of the ensemble of classifiers is removed and a new classifier is added to the ensemble. Finally CSC in Weka is a cost-sensitive learning algorithm, in which two methods can be used to introduce cost-sensitivity: reweighting training instances according to the total cost assigned to each class; or predicting the class with minimum expected misclassification cost, rather than the most likely class. It is a meta classifier that makes its base classifier cost-sensitive and is suitable for processing the unbalanced data set in a traditional batch learning. Incremental learning algorithms for unbalanced data are not

included in the MOA platform. So we selected CSC to test in the study, which is valid for unbalanced data in batch learning setting.

3.3. The evaluation measure for data streams

In batch learning, the evaluation measure allows training data and test data to be selected for constructing the models and then performs the tests repeatedly. With the additional data, such as in the case of a data stream, which may be continuously generated, repeat training and testing are impossible. We have to complete the performance measure by reducing the numbers of folds in a limited time and memory for stream data. Bifet et al. proposed to compare several evaluation methodologies. Two main approaches [1], [5] are used for building a picture of accuracy over time. When traditional batch learning reaches a scale where cross validation is too time consuming, it is often acceptable to instead measure performance on a single holdout set. This is most useful when the division between train and test sets has been predefined, so that results from different studies can be directly compared. In testing data stream models a common approach is the Interleaved Test-Then-Train or Prequential method. Each individual example can be used to test the model before it is used for training, and from this the accuracy can be incrementally updated. When intentionally performed in this order, the model is always being tested on examples it has not seen. This scheme has the advantage that no holdout set is needed for testing, therefore making maximum use of the available data. It also ensures a smooth plot of accuracy over time, as each individual example will become increasingly less significant to the overall average.

In a data streaming setting, the most common evaluation measure for data stream is Prequential accuracy [11]. Bifet et al. [7] stated that Kappa statistic [11], which is used in Prequential method, has advantages over the traditional accuracy measures when data streams have evolving unbalanced labels. Kappa statistic is also better than traditional measures such as the area under the ROC curve [6].

4. The experiment design and results from computer simulation

This study used Weka for testing the classifiers in the batch and data stream setting. Weka classifiers may be either incremental or nonincremental [20]. We designed four scenarios for testing our classifiers. In Scenario 1, we used NB, VFDT, OBA and CSC for batching learning. NB, VFDT, OBA are incremental algorithms and CSC is a nonincremental algorithm.

5000 records were used in the simulation. Table 1 shows that the overall classification accuracies for CSC, NB, VFDT, and OBA are 71.5%, 67.1%, 67.0, and 62.6%. The ROC values are 0.699, 0.699, 0.638, and 0.655. Figure 2 shows the ROC charts for the four methods in batch learning. It shows that CSC and NB are the best classification algorithms in the overall performance category, as they have the biggest areas under the ROC curve. OBA is the worst and VFDT is in the third place.

To compare with batch learning, we performed a Holdout evaluation and prequential evaluation in a data stream setting. In Scenario 2, the Holdout evaluation was applied. The data used for testing and training are a stream of 5000 instances. We set parameters as Testsize=50, Trainsize=500, and SampleFrequency=1 in the simulation. Table 2 presents the mean classification accuracy rates in the Holdout evaluation. The rates for NB, VFDT, OBA and CSC amount to 56.4%, 63.7%, 58.5% and 65.6%, respectively. VFDT and CSC have better mean accuracies with Holdout evaluation. Figure 3 is the learning curve for this stream in the Holdout measure. In Figure 3 we can see the 100 (100=5000/50) accuracy values in the Holdout evaluation and the plot of accuracy is not smooth. Some accuracy values with VFDT and CSC between the 30^{th} instance and the 80^{th} instance in Figure 3 are high. The unstable results are difficult to represent in terms of performance in a data stream setting.

Table 1. Classification results for the four algorithms used in Scenario 1

Models	ROC	Overall	Non-human factor cases	Human factor cases
		(%)	(%)	(%)
NB	0.699	67.1	72.4	57.1
VFDT	0.638	67.0	86.9	29.0
OBA	0.655	62.6	64.31	52.1
CSC	0.699	71.5	75.0	64.9

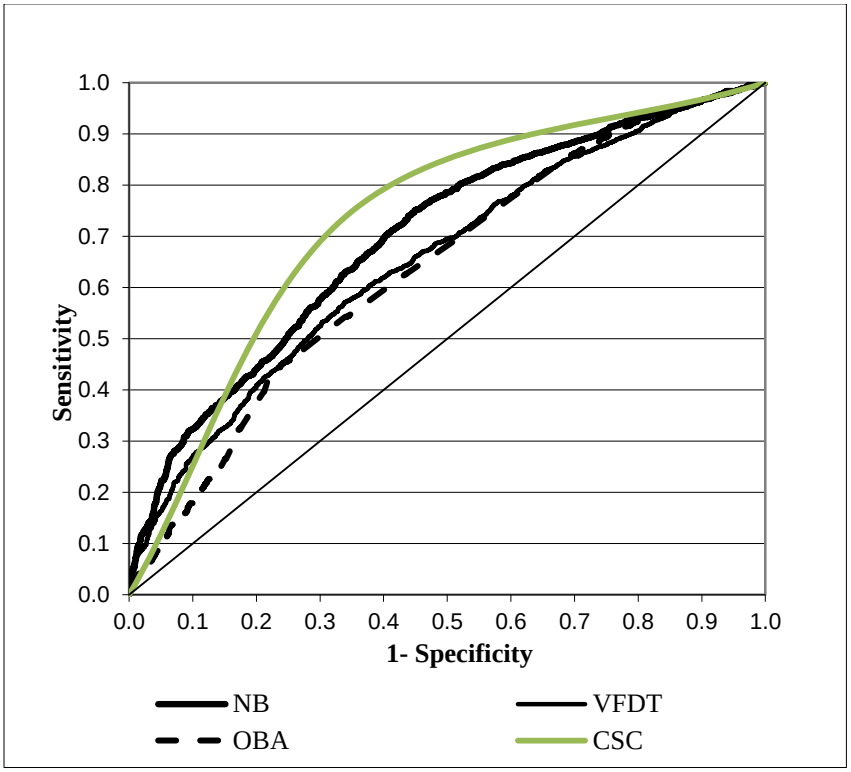

Figure 2. The ROC charts results for the four algorithms in Scenario 1

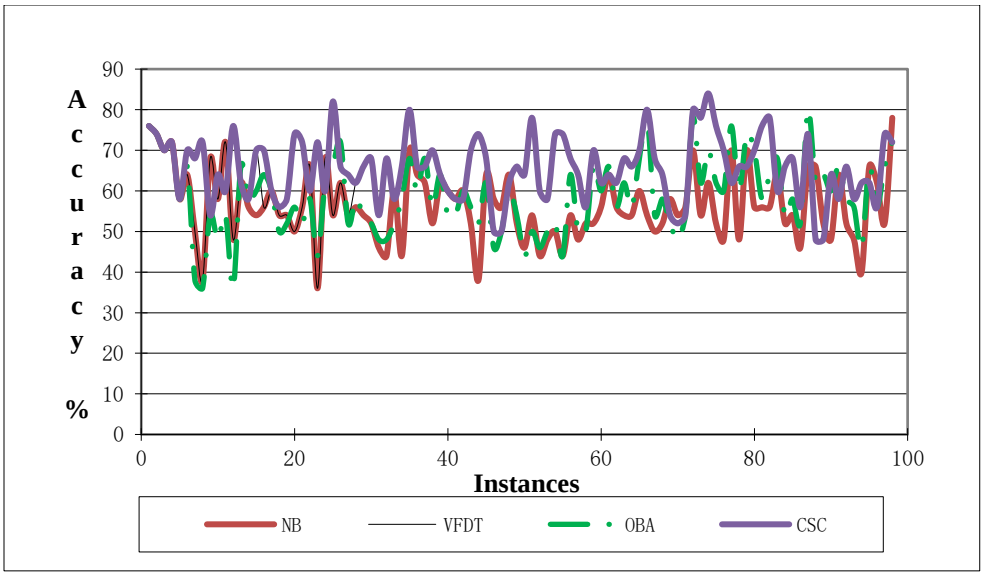

Figure 3. The classification accuracy rates for Holdout in Scenario 2

Table 2. Mean accuracy rates for Holdout in Scenario 2

Classifier	Mean Overall Accuracy (%)
NB	56.4
VFDT	63.7
OBA	58.5
CSC	65.6

In Scenario 3, we performed a prequential evaluation, testing and then training, using data streams of 5000 instances. NB, VFDT, OBA and CSC were used in the scenario. Window size was set to 100. Figure 4 provides the learning curve for prequential accuracy and Table 3 reports the total prequential accuracy rates with window size of 100. Figure 4 suggests that the plots with NB, VFDT and OBA included many fluctuations. The curve with CSC is smooth. CSC produced the best performance in prequential accuracy. The result shows that in order to obtain steady results, bigger window sizes should be set.

In Scenario 4, we performed a prequential evaluation, testing and then training, using data streams of 5000 instances, with a window size of 1000. NB, VFDT, OBA, and CSC were used in this scenario. Figure 5 shows the learning curve for prequential accuracy. We can see that the plot with a window size of 1000 is smooth. Table 4 reports the total accuracy, human factor accuracy, and non-human factor accuracy. Although Figure 5 shows that NB produced the worst performances in measuring prequential accuracy, the mean human factors accuracies of NB and CSC with 66.5%, 70.9% in Table 4 are greatly better than VFDT and OBA. Those of VFDT and OBA are very low with 14.7%, 16.8%. CSC has the best human factor accuracy in all the algorithms. Perhaps this result is due to the fact that CSC is suitable for unbalanced data. It is a meta classifier with SVM (Support Vector Machine) algorithm being used as the base classifier. In this algorithm, the cost matrix was manually adjusted to [0.3, 1.0], since about 1/3 of the incidents has been classified by human experts as caused by human related factors and the rest by non-human factors.

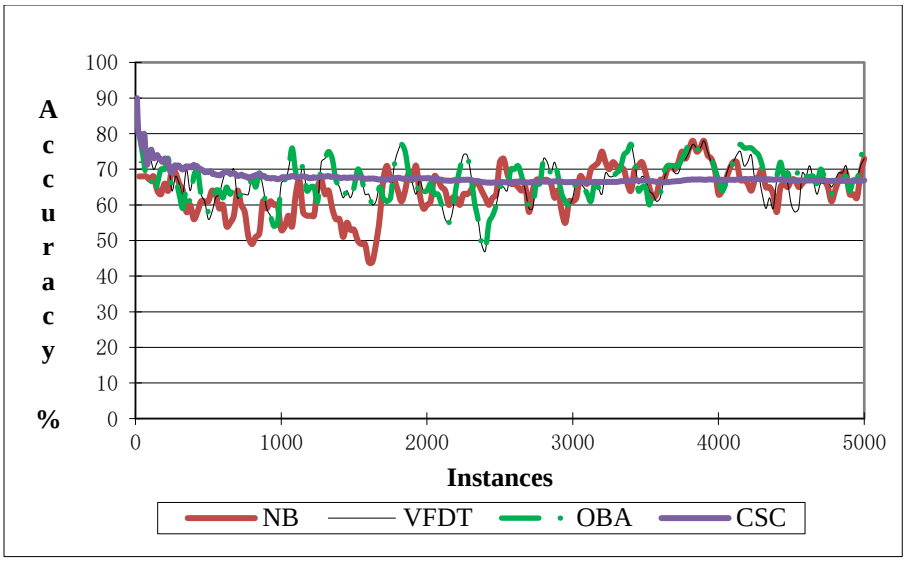

Figure 4. Prequential accuracy rates for the sliding window size of 100 in Sceniario 3

Table 3. Prequential mean accuracy rates for the sliding window size of 100 in Sceniario 3

Classifier	Mean Overall Accuracy (%)
NB	64.04
VFDT	66.54
OBA	66.75
CSC	67.66

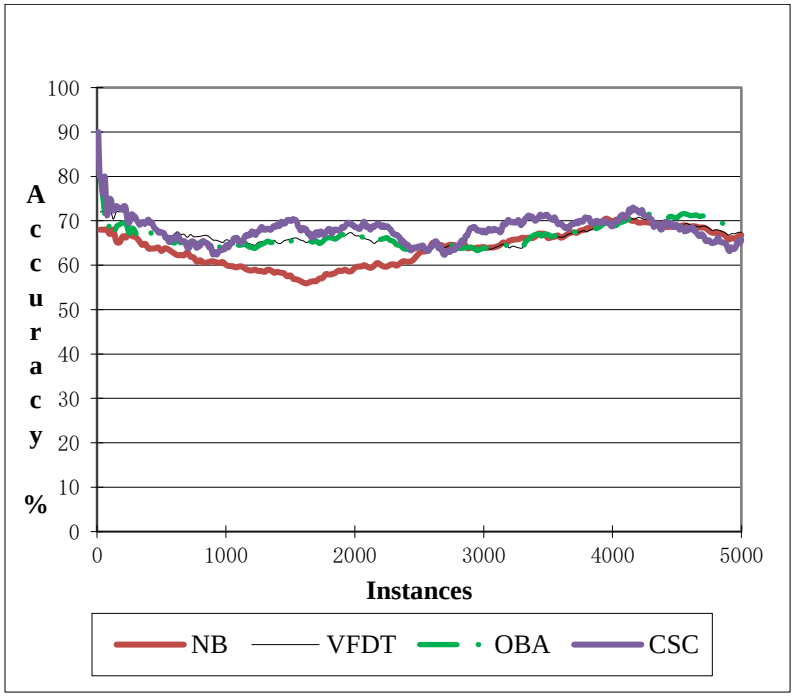

Figure 5. Prequential accuracy rates for the sliding window size of 1000 in Sceniario 4

Table 4. Prequential mean accuracy rates for the sliding window size of 1000 in Sceniario 4

Classifier	Mean Overall Accuracy Rates [%]	Non-human factor cases [%)]	Human factor cases [%]
NB	63.93	62.8	66.5
VFDT	66.75	93.8	14.7
OBA	66.65	93.01	16.8
CSC	61.0	53.1	70.9

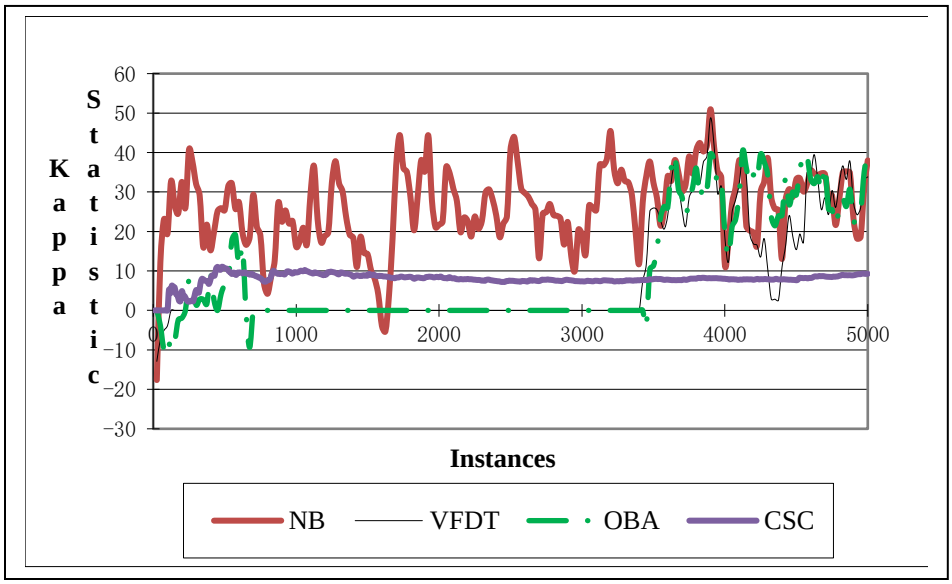

Figure 6. Prequential Kappa statistic for the sliding window size of 100 in Sceniario 3

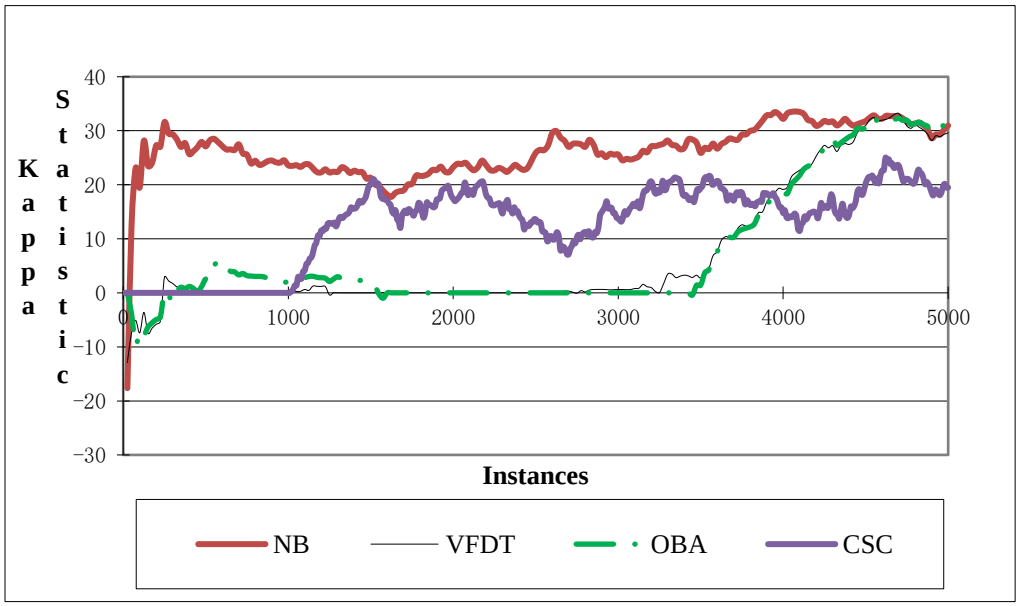

Figure 7. Prequential Kappa statistic for the sliding window size of 1000 in Sceniario 4

Table 5. Prequential mean Kappa statistic for the sliding window size of 100 in Scenario 3

Classifier	Mean Kappa
NB	26.23
VFDT	8.05
OBA	9.40
CSC	7.93

Table 6. Prequnetial mean Kappa statistic for the sliding window size of 1000 in Scenario 4

Classifier	Mean Kappa
NB	26.31
VFDT	6.93
OBA	7.78
CSC	17.43

The Kappa statistic, which normalizes a classifier's accuracy by a chance predictor, is an appropriate measure in data stream mining due to potential changes in the class distribution. Figure 6 represents Kappa statistic plots using prequential measure with a sliding window size of 100 in Scenario 3. Figure 7 shows Kappa statistic charts of prequential measure with a sliding window size of 1000 in Scenario 4. In the Kappa statistic charts, we can also see that the plot with a size of 1000 is smoothed. The plot with a size of 100 includes many fluctuations. It shows that NB has the best performance, much better than the other three algorithms. Table 5 shows the Kappa statistics with a sliding window size of 100 in Scenario 3. Table 6 shows the Kappa statistics for a window size of 1000 in Scenario 4. It verifies that NB produced the best performance compared to the other three algorithms.

5. Conclusion

In the study, we identified the presence of human factors from aviation incidents using data stream models. Topic mining was used to extract the structured information from the textual data. Then four different data stream algorithms were tested to assess their potential in classifying the incidents. Our results show that NB is the best classification algorithm. Our results are significant because aviation incidents data stream is continuous and our study demonstrates the potential of data stream models in classifying these incidents. We conclude that NB and CSC are the best classification algorithms by ROC charts. However, they cannot predict the incident with human factor/non-human factor in real-time. We use two evaluation measures in the data stream setting. One is the Holdout, and the other is the Interleaved Test-Then-Train or Prequential. In the Holdout, VFDT and CSC are similar in accuracy, and both are better than NB and OBA. The prequential measure can solve the problem, predicting the class label in real time. In the Accuracy curve of sliding window prequential with size 100, CSC is better than other algorithms. But the plot with a size of 100 included many fluctuations. The plot with a size of 1000 is smoothed. The prequential accuracy rates with the sliding window size of 1000 show that NB and CSC are better in classifying human factor related incidents than the other two algorithms. Kappa statistic charts of prequential measure with the sliding window size of 1000 shows that NB has the best performance and is significantly better than the other algorithms. For the unbalanced data stream, Kappa statistic is better to describe the performance of the classification algorithms. The interesting thing is that ROC in batch learning and prequential Kappa measure both conclude that NB is the best classification model. It suggests that we may use the measure of the traditional batch learning to scale the performance of the data stream. We can split the data stream into several small datasets. Batch learning is used on the segmented data. It provides us an alternative way to process the data stream. In addition, CSC is a valid algorithm for unbalanced data in a batch learning setting, but it is not the best in the kappa statistic for data stream. It could be that we need to develop a new incremental algorithm for the data stream.

Acknowledgment

This work was supported by (1) Anhui Provincial Natural Science Foundation of China (1508085MF114). (2) Technology Foundation for Selected Overseas Chinese Scholar (2014).(3) Anhui Provincial Science Foundation for Youths (1508085QF137).

References

[1] Bifet, G. Holmes, R. Kirkby, and B. Pfahringer. MOA: Massive Online Analysis http://sourceforge.net/projects/moa-datastream/. Journal of Machine Learning Research (JMLR), 11, pp. 1601-1604, 2010.

[2] Andrzejczak, C., Karwowski, W., & Mikusinski, P. Application of diffusion maps to identify human factors of self-reported anomalies in aviation. Work. A Journal of Prevention, Assessment and Rehabilitation, 41, pp.188-197, 2012.

[3] Cano, Ignacio, and Muhammad Raza Khan. "ASML: Automatic Streaming Machine Learning." http://homes.cs.washington.edu/~icano/projects/asml.pdf

[4] Barrientos, F., Castle, J., McIntosh, D., & Srivastava, A., Preliminary Evaluation of an Aviation Safety Thesaurus' Utility for Enhancing Automated Processing of Incident Reports,2007.http://ntrs.nasa.gov/search.jsp?R=20070025054, NASA Technical Reports Server(NTRS)

[5] Bifet, Albert, Richard Kirkby, Philipp Kranen, and Peter Reutemann. "Massive online analysis manual." University of Waikato, New Zealand: Centre for Open Software Innovation, 2009.

[6] Bifet, Albert, and Eibe Frank. "Sentiment knowledge discovery in twitter streaming data." In Discovery Science, Springer Berlin Heidelberg, pp. 1-15, 2010.

[7] Bifet, Albert, Gianmarco de Francisci Morales, Jesse Read, Geoff Holmes, and Bernhard Pfahringer. "Efficient Online Evaluation of Big Data Stream Classifiers." In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, pp. 59-68, 2015.

[8] Ding, Shifei, Fulin Wu, Jun Qian, Hongjie Jia, and Fengxiang Jin. "Research on data stream clustering algorithms." Artificial Intelligence Review, 43(4), pp. 593-600, 2015.

[9] Domingos, Pedro, and Geoff Hulten. "Mining high-speed data streams." In Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp. 71-80, 2000.

[10] Gama, João, Raquel Sebastião, and Pedro Pereira Rodrigues. "Issues in evaluation of stream learning algorithms." In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp. 329-338, 2009.

[11] J. Cohen. A coecient of agreement for nominal scales. Educational and Psycho-logical Measurement, 20(1):37-46, April 1960.

[12] Telec, Zbigniew, Tadeusz Lasota, Bogdan Trawiński, and Grzegorz Trawiński. "An Analysis of Change Trends by Predicting from a Data Stream Using Neural Networks." In Flexible Query Answering Systems, Springer Berlin Heidelberg, pp. 589-600, 2013.

[13] Reason, James. Human error. Cambridge university press, 1990.

[14] O'Hare, D. Cognitive Functions and Performance Shaping Factors in Aviation Accidents and Incidents. The International Journal of Aviation Psychology, 16(2), pp. 145-156, 2006.

[15] Péladeau, N., & Stovall, C., Application of Provalis Research Corp.'s statistical content analysis text mining to airline safety reports. Flight Safety Foundation Web site:<http://www.flightsafety.org/gain/Provalis_text_mining_report. pdf>(retrieved 19.08. 2005).

[16] Posse, C., Matzke, B., Anderson, C., Brothers, A., Matzke, M., & Ferryman, T. (2005). Extracting information from narratives: An application to aviation safety reports. In the Proceedings of the IEEE 2005 Aerospace Conference, pp.3678-3690, 2005.

[17] Shi, Donghui, Jian Guan, and Jozef Zurada. "Cost-Sensitive Learning for Imbalanced Bad Debt Datasets in Healthcare Industry." In Computer Aided System Engineering (APCASE), 2015 Asia-Pacific Conference on, IEEE, pp. 30-35, 2015.

[18] Wiener, E. L., & Nagel, D. C., Human factors in aviation. Gulf Professional Publishing, 1998.

[19] Weiss, Gary M., Kate McCarthy, and Bibi Zabar. "Cost-sensitive learning vs. sampling: Which is best for handling unbalanced classes with unequal error costs?" Int .Conf. on Data Mining, pp. 35-41., 2007.

[20] Witten, I. H., E. Frank, and M. A. Hall. Data Mining: Practical Machine Learning Tools and Techniques, 3rd edn, Morgan Kaufmann, Burlington, MA, 2011.